

Design and implementation of Human-computer interaction geological modeling software based on Qt framework

Jie Xiong
School of Elect. and Info.
Yangtze University
Jingzhou, China

Shenhan Shi
School of Elect. and Info.
Yangtze University
Jingzhou, China

Fangwen Wang
School of Elect. and Info.
Yangtze University
Jingzhou, China

Abstract: Three-dimensional geological modeling is the basis of exploration and exploitation of oil, gas, and solid mineral resources. It has vital significance for engineering decision-making and scientific management. A cross-platform geological modeling and visualization software with independent intellectual property rights is developed, based on the Qt framework. The 2.5D geological model is built with the Human-computer interaction (HCI) method, and converted to 3D model which displayed real time based on OpenGL. Software testing results show that the software can run on Windows, Linux platforms and meeting the basis requirement of cross-platform HCI modeling of the mineral exploration.

Keywords: 2.5D geological modeling; human-computer interaction; software design; Qt; cross-platform; 3D visualization.

1. INTRODUCTION

Three dimensional geological modeling was first proposed by Simon W Houlding in 1994 [1]. With its continuous development, Three dimensional geological modeling have been widely used in the field of geological exploration, mathematical geology, geophysics, mine surveying, mine geology, graphics, scientific visualization, and engineering applications [2-5]. it is the basis for the exploration and development of oil and gas resources, and solid mineral resources. It is an effective way to reproduce 3D information and comprehensive analysis of geological bodies, by introduction of 3D geological modeling technology into geological body analysis system.

The first 3D visualization modeling software of geological body was born in the Western Countries. With its continuous development, 3D geological modeling has been widely employed in various fields abroad. Since 1980s, several

successful 3D modeling software have been developed, such as: EarthVision [6], GOCAD [7], LYNX [8], Vulcan [9], EarthCube [10]. In China, the research of 3D visualization technology started in the early 90s, and the software has been developed less[4]. Although there are some excellent software such as RDMS, SLGRAPH, GeoView [11] and so on, but in general, there is still a big gap with the advanced level of foreign countries. Most of the geological modeling software used in China are imported from abroad. It is necessary to develop cross platform 3D geological modeling and visualization software with independent intellectual property rights to meet the needs of solid mineral exploration and exploration and development of oil and gas resources.

2. SOFTWARE DESIGN

2.1 General System Diagram

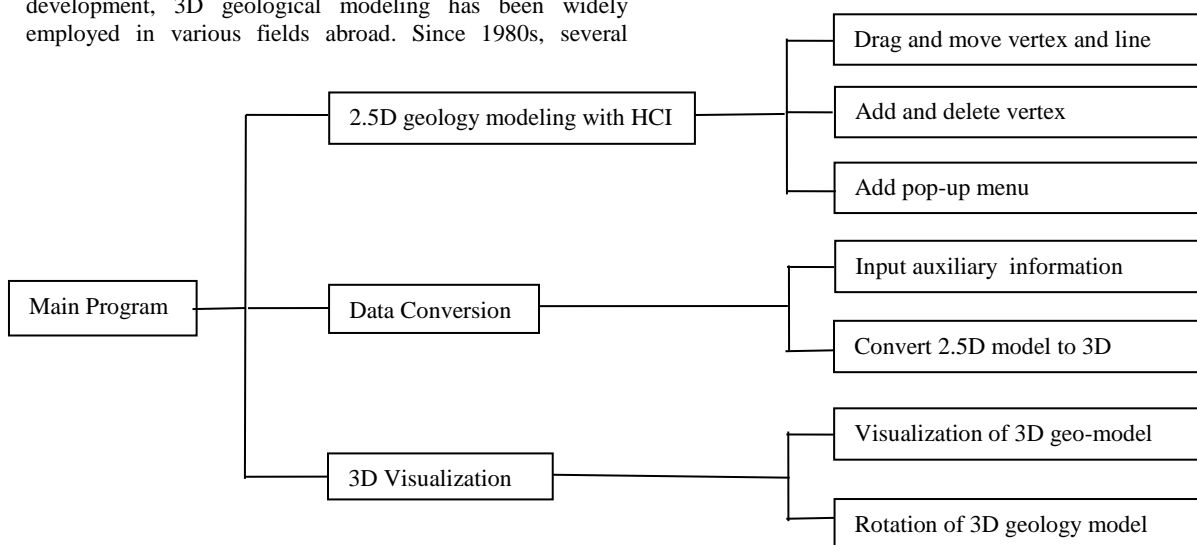


Figure 1. General System Diagram

The software consists of a main program and three sub modules: 2.5 dimensional geological modeling module with human-computer interaction, data conversion module, and 3D visualization module. The software function module diagram is shown in Figure 1. The interactive 2.5 dimensional geological modeling module is to modeling the ore-body using HCI; The data conversion module is to convert 2.5D model to 3D model using the auxiliary information; The 3D visualization module is to visualize the 3D geological model.

2.2 Main Module Design

2.2.1 HCI modeling module

The software start with a Qt GUI window, which contains an interactive graphics widget, a 3D model rendering widget, and several auxiliary input widgets.

For the 2.5 dimensional interactive modeling module, the user can draw geological model profile in the rendering window, and modify the profile (adding or deleting vertex, selection, drag and drop vertex and line). The module flow chart is shown in figure 2.

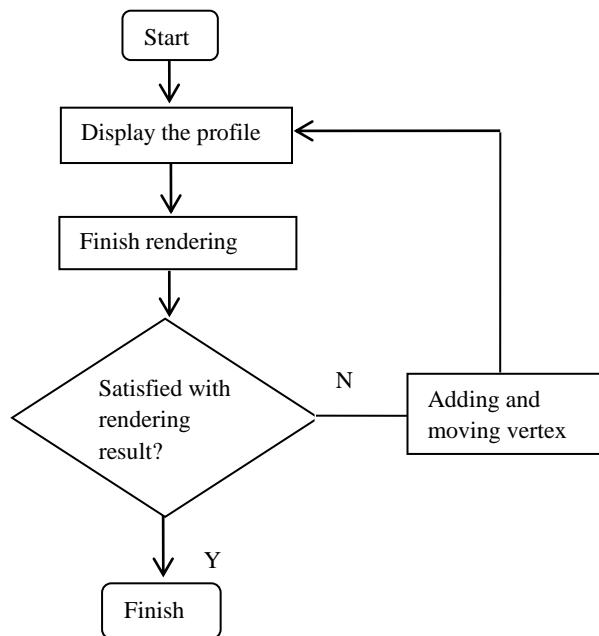


Figure 2. Flow chart of 2.5D modeling

2.2.2 Three dimensional visualization

We can draw the surface of geological model after the real 3D coordinates are calculated from the profile and the auxiliary information, and rotate the model by rotation function in OpenGL. If we do not satisfied with the 3D model, we can modify it by modeling the profile and auxiliary information. The flow chart is illustrated in Figure 3.

2.2.2 Data conversion

It is important to convert the 2D coordinates to 3D coordinates, since displaying 3D model need real 3D coordinates. The conversion theory described as following:

Supposing the dip of geological model $I = 0$, the top view of geological model is shown in Figure 4(a). We can calculate the X_1, Y_1 and X_4, Y_4 , using the information $x_0, y_0, W/L, A$

$$x_1 = x_0 - L / 2 \times \sin(A) - W / 2 \times \cos(A) \quad (1)$$

$$y_1 = y_0 - L / 2 \times \cos(A) + W / 2 \times \sin(A) \quad (2)$$

$$x_4 = x_0 + L / 2 \times \sin(A) - W / 2 \times \cos(A) \quad (3)$$

$$y_4 = y_0 + L / 2 \times \cos(A) + W / 2 \times \sin(A) \quad (4)$$

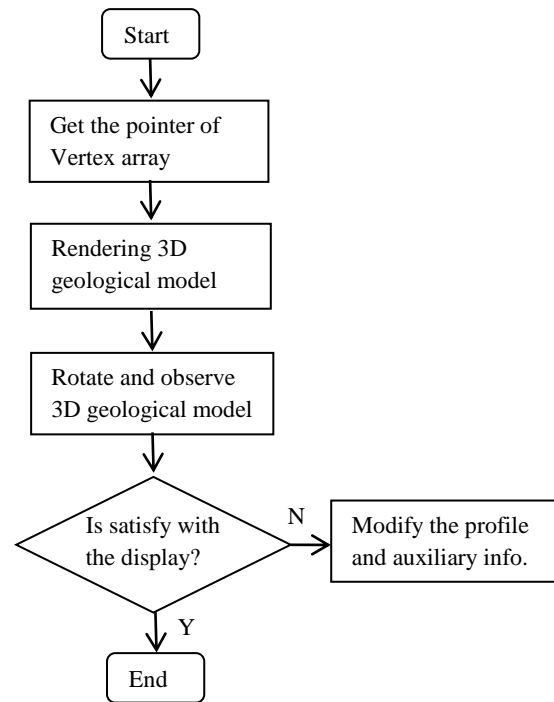


Figure 3. Flow chart of 3D visualization

Assuming that the profile of the geological model is shown in Figure 4 (b), the coordinate of A is (x_1, y_1) , and the difference between the abscissa of A and the C is W. Combining the top view with the profile, you can get Figure 4 (c), which calculates the coordinates (x, y) of each vertex according to the linear scale. The coordinates Z can be calculated based on the plotted profile and the actual proportion of the drawing, plus the depth of the embedded surface. Then, the coordinates of all the vertex of the front and back profiles are computed by the loop.

If I is not equal to 0, that is, the geological model rotates about the axis of rotation I . The diagram of vertex rotation is shown in Figure 4 (d). L_0 is the distance from point to line:

$$dx = (L_0 - L_0 \cos(I)) \sin(I) \quad (5)$$

$$dy = (L_0 - L_0 \cos(I)) \cos(I) \quad (6)$$

$$dz = L_0 \sin(\alpha + I) - L_0 \sin(\alpha) \quad (7)$$

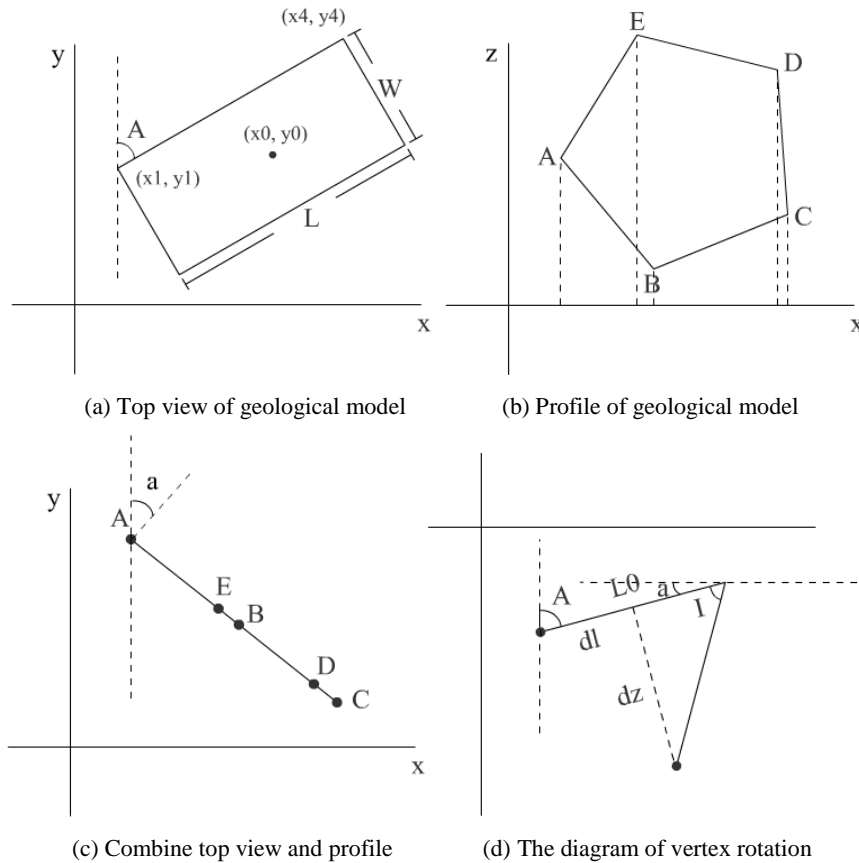


Figure 4. Data conversion module

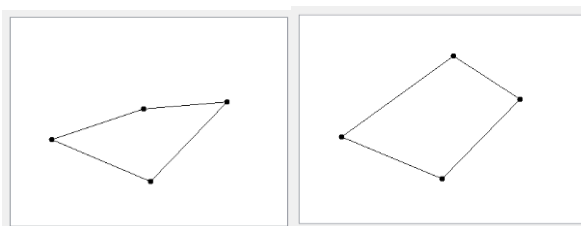
3. SOFTWARE IMPLEMENT

3.1 2.5-D modeling with HCI

To draw the profile of geological model, we must implement the functions such as dynamic drawing line, selecting and dragging the vertex and line, adding and deleting the vertex, and the pop-up menus.

Dynamic drawing lines: it is implemented by using the QPainter class and double buffering technology in Qt.

Selecting and dragging the vertex and line: It is implemented by drag and drop of mouse. When the vertex or line is selected, it turns to read. After you select the point, press and hold the left mouse button, move the mouse, you can drag this point, and this point and other points will change with the line. The effect of drag and drop of vertex is shown in Figure 5:



(a) Before dragging vertex (b) After dragging vertex

Figure 5. Illustration of dragging a vertex by mouse drag and drop

Pop-up menu: It is supported by QWidget and its sub-classes and can be called directly.

Adding and deleting the vertex: It is implemented by calling addPoint() and deletePoint() function respectively. The addPoint() function is listed in following:

```
void PaintArea::addPoint(QImage &theImage)
{
    if(choose_line_index >= 0)
    {
        QPainter pp(&theImage);
        pp.setPen(QPen(Qt::white, 1));
        if (choose_line_index < pvec.size()-1)
        {
            pp.drawLine(pvec[choose_line_index], pvec[choose_line_index+1]);
            pvec.insert(&pvec[choose_line_index] + 1, theAddPoint);
            pp.setPen(QPen(Qt::black, 1));
            pp.drawLine(pvec[choose_line_index], pvec[choose_line_index+1]);
            pp.drawLine(pvec[choose_line_index+1], pvec[choose_line_index+2]);
        }
    }
}
```

```

else
{
    pp.drawLine(pvec[0], pvec[pvec.size()-1]);
    pvec.push_back(theAddPoint);
    pp.setPen(QPen(Qt::black, 1));
    pp.drawLine(pvec[pvec.size()-1], pvec[pvec.size()-2]);
    pp.drawLine(pvec[pvec.size()-1], pvec[0]);
}
pp.setPen(QPen(Qt::black, 4));
pp.setBrush(Qt::black);
pp.drawEllipse(theAddPoint, 2, 2);
update();
}
}

```

3.2 Data conversion

All the vertices of the polygonal profile are stored in an array. After the profile of geological model and other auxiliary information of it are obtained, we can transform the 2.5 dimensional geological model to a three-dimensional geological model, using the data conversion algorithm described in section 2.2.2.

3.3 Three Dimensional Visualization

After obtaining the spatial coordinates of the three-dimensional model of the geological model, the 3D model is rendered by OpenGL. We use loops to draw the outer face of the geological model. We define a two-dimensional array that stores colors, which allow us to draw outer face with different colors. The rotation of the geological model is implemented by the rotation function in OpenGL, and it is associated with the mouse event to implement rotation with mouse. The codes for rendering the 3D mode are listed as follow:

```

//Rendering two faces
glBegin(GL_POLYGON);
glColor3f(0.0f,1.0f,0.0f);
index = 0;
while(index < fqvec_3D.size())
glVertex3fv(fvertices[index++]);
glEnd();
//Rendering the sides
index = 0;
while(index < fqvec_3D.size()-1)
{
glBegin(GL_POLYGON);
glColor3f(0.5, 0.5, 0.5);
glVertex3fv(fvertices[index]);

```

4. SOFTWARE TESTING RESULTS

The software testing platform is Intel i5 -3470 3.2GHz CPU, 4GB memory, We compile and run the software on the

Windows and Linux operating systems respectively. The results are shown in Figure 6 and figure 7.

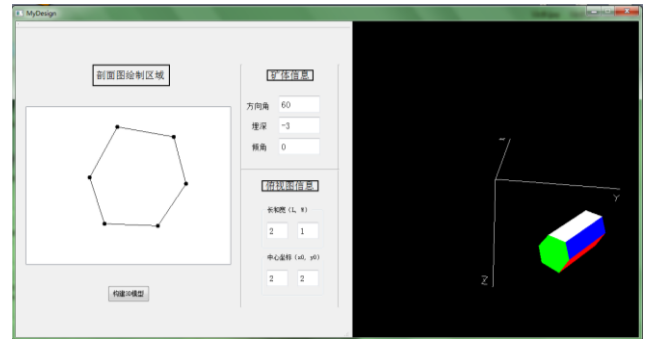


Figure 6. Testing result running on Windows

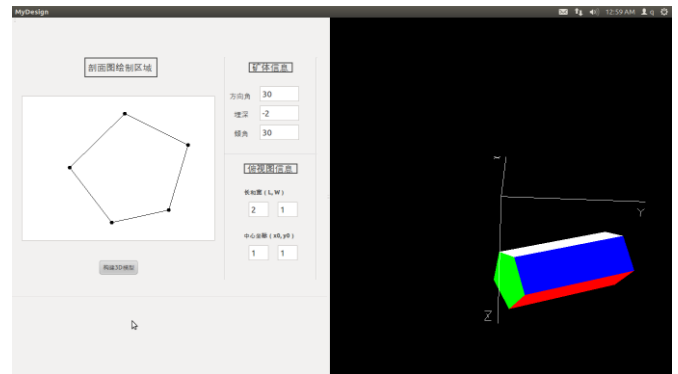


Figure 7. Testing result running on Linux

5. CONCLUSION

A 2.5 dimensional modeling software with human computer interaction is developed based on Qt software framework and OpenGL graphics library. The software can be used to model the 2.5 dimensional geological model by human-computer interaction. After the 2.5 dimensional model is transformed into a three-dimensional model, the 3D geological model is displayed in real time using OpenGL. The software test results show that the software, which can run on both Windows and Linux, can be employed for human computer interaction modeling in the field the mineral exploration.

6. ACKNOWLEDGMENTS

This work is support by the National Science Foundation of China (No. 61273179, No. 61673006), and Science and Technology Research Project of Education Department of Hubei Province of China (No. D20131206, No. B2016034, No. 20141304).

7. REFERENCES

- [1] Houling, S.W. 1994. 3D geoscientific modeling--computer technique for geological characterization, Springer-Verlog.
- [2] Wang, Z. Qu, H. Wu, Z. 2016, Formal representation of 3D structural geological models, Computers & Geosciences, Vol. 90, Part A, 10-23.
- [3] Maesano, F. D'Ambrogi, C. 2017. Vel-IO 3D: A tool for 3D velocity model construction, optimization and time-depth conversion in 3D geological modeling workflow, Computers & Geosciences, Vol. 99, 171-182.

- [4] Muzik, J. Vondrackova, T Sitanyiova, D. et al. 2015. Creation of 3D Geological Models Using Interpolation Methods for Numerical Modelling, *Procedia Earth and Planetary Science*, Vol.15, 25-30.
- [5] Wang, G. Zhu, Y. Zhang, S. et al. 2012, 3D geological modeling based on gravitational and magnetic data inversion in the Luanchuan ore region, Henan Province, China, *Journal of Applied Geophysics*, Vol.80, 1-11.
- [6] <http://www.dgi.com/earthvision/evmain.html> (last accessed on July 15, 2017).
- [7] <http://www.pdgm.com/products/gocad/> (last accessed on July 15, 2017).
- [8] www.lynxgeo.com/HTML/TechPapers/lynx.htm (last accessed on July 15, 2017).
- [9] <http://www.maptek.com/products/vulcan> (last accessed on July 15, 2017).
- [10] <https://www.earthcube.org/> (last accessed on July 15, 2017).
- [11] <http://www.geoview.com.cn/> (last accessed on July 15, 2017).